Computers Math. Applic. Vol. 26, No. 6, pp. 13–31, 1993
Printed in Great Britain.
0898-1221/93 \$6.00 + 0.00

The Cubic Algorithm for Global Games with Application to Pursuit-Evasion Games

E. A. GALPERIN
Département de mathématiques et d'informatique, Université du Québec à Montréal
C.P. 8888, Succ. "A", Montréal, Québec H3C 3P8, Canada

Abstract—New algorithms for solution of nonconvex global games defined over a cube are presented. Application to differential games and to pursuit-evasion games is considered, and extension is made for games over general compact and robust sets defined by constraints that may depend on time and on state variables. The methods are deterministic and monotonically set-convergent to deliver, in the limit, the unique exact full globally optimal minimax and maximin solutions for both players, or the entire global saddle set, if it exists. A stopping rule is provided for determining approximate solutions with a given precision in a finite number of iterations. A modification is considered to play on mistakes of the opponent. On-line replacement of cost functionals is possible in the course of iterations, according to changing interests of the players. No separability nor convexity-concavity assumptions are imposed on the cost functional, and variational methods are not used. The ideas are illustrated by examples and application is made to determining the globally optimal closed-loop strategies for the ship-torpedo collision-avoidance differential game with manoeuvrability constraints.

1. INTRODUCTION

The cubic algorithm [1,2] finds the global minimum value

$$s^\circ = \min_{x \in \bar{C}} f(x), \quad \bar{C} \subset \mathbb{R}^n \tag{1.1}$$

and the set of all global minimizers

$$\bar{K}^\circ = \{x \in \bar{C} | f(x) = s^\circ\} \tag{1.2}$$

for a Lipschitz continuous function $f(x)$:

$$|f(x) - f(x')| \le L||x - x'||, \quad L = \text{const.} > 0,\ x, x' \in \bar{C}, \tag{1.3}$$

minimized over a closed cube $\bar{C} \subset \mathbb{R}^n$.

The algorithm can be briefly described as follows.

Consider the partition of the cube $\bar{C}$ into 2^n equal closed subcubes $\bar{C}_i^1, i = 1, \dots, 2^n$, with edges of the length $c/2, c =$ const., and take as their representatives the 2^n peaks $x_i = \{x_i^1, \dots, x_i^n\}$, $x_i^j = 0$ or $c/2, i = 1, \dots, 2^n$. The diameter of each $\bar{C}_i^1$ is $d(C_i^1) = \max ||x - x'|| = c\sqrt{n}/2, x, x' \in \bar{C}_i^1$. For simplicity, $\bar{c}$ is considered axes oriented with a peak at the origin.

Deletion constants: $r_m = Ld(C_i^m) = Lc\sqrt{n}/2^m, m = 1, 2, \dots; r_m \to 0$ as $m \to \infty$.

Deletion operator. Given s_0 as a comparison constant, delete all $\bar{C}_i^1$ for which $f(x_i) - s_0 > r_1$.

The Algorithm. Compute $s_0 = f(0)$ for the peak $x_0 = 0$. Make the first partition $\bar{C} = \cup \bar{C}_i^1$ and the first deletion. For the remaining C_i^1 define $I_1 = \{i | f(x_i) - s_0 \le r_1, x_i \in \bar{C}\}$ and the closure $\bar{K}_1 = \{x | x \in \bar{C}_i^1,\ i \in I_1\}$. Compute $s_1 = \min f(x_i), i \in I_1$; clearly $s_1 \le s_0$. Partition each of the $\bar{C}_i^1 \subset \bar{K}_1$ in the same way and repeat the iteration replacing s_0, r_1 by s_1, r_2, etc.

This work was supported by the Natural Sciences and Engineering Research Council of Canada, Grant No. A3492. Written in Montreal, January-February 1992.

Typeset by AMS-TeX

THEOREM [2, pp. 638–639]. *We have*

$$\lim s_m = s^\circ,\ \lim \bar{K}_m = \bar{K}^\circ \text{ as } m \to \infty.$$

Here, we shall modify the algorithm to solve nonconvex game problems of finding the global minimax and maximin values

$$a^\circ = \min_{u\in U}\max_{v\in V} f(u,v), \quad a^* = \max_{v\in V}\min_{u\in U} f(u,v) \tag{1.4}$$

and the entire sets

$$A^\circ, \quad A^* \tag{1.5}$$

of elements $(u,v) \in U \times V$ that deliver the global minimax a° and maximin a^* values respectively.

The sets $U \subset \mathbb{R}^{n_1}$, $V \subset \mathbb{R}^{n_2}$ are supposed to be compact and the function $f(u,v)$ to be Lipschitz continuous on $U \times V$ with a known constant $L > 0$. In the sequel, we shall identify $(u,v) = x \in U \times V \subset \mathbb{R}^n, n = n_1 + n_2$, and $f(u,v) \equiv f(x)$; then the Lipschitz condition for $f(u,v)$ on $U \times V$ is given by (1.3) with $x, x' \in U \times V$. If we consider subcubes $\bar{C}_i \subset U \times V$, then within each $\bar{C}_i$ we can use better estimates than (1.3):

$$|f(x) - f(x')| \le L_i\|x - x'\|, \quad 0 < L_i \le L,\ x, x' \in \bar{C}_i. \tag{1.6}$$

If we take x' in the center $x_i \in \bar{C}_i$, then from (1.6) we obtain a bound on the variation of $f(x)$ *vis-à-vis* x_i within $\bar{C}_i$:

$$|f(x) - f(x_i)| \le L_i\|x - x_i\| \le L_i \max_{x\in\bar{C}_i} \|x - x_i\| = \frac{1}{2} L_i\, d_i = \frac{1}{2} L_i\, c_i\sqrt{n}, x \in \bar{C}_i, \tag{1.7}$$

where d_i is diagonal (diameter) of $\bar{C}_i$ and c_i is its edge.

We do not impose convexity or concavity assumptions, nor differentiability of the function $f(u,v)$ or the boundaries $\partial U, \partial V$. We do not assume the existence of a saddle point, thus, in general, $a^\circ \ne a^*, A^\circ \ne A^*$. Such game problems for which the full global optimal solution (1.4)–(1.5) is sought are called global games, the notion introduced in [3,4]. For compact U, V and continuous $f(u,v)$, the sets A°, A^* of (1.5) are both nonempty, see, e.g., [3, pp. 8–10], or [4, pp. 133–135]. Thus, the values a°, a^* of a global game are well-defined, together with the exact sets A°, A^* of elements in $U \times V$ which deliver those values.

It is worth noting that, if gradient or variational methods are applied, then min-max and max-min in (1.4) become relative (i.e., local) and in a non-convex-concave game problem there may exist many different local values $\tilde{a}^\circ$, $\tilde{a}^*$ with corresponding sets $\tilde{A}^\circ$, $\tilde{A}^*$. Such local solutions different from $a^\circ, a^*, A^\circ, A^*$ of (1.4), (1.5) are not considered here. In contrast, the global values a°, a^* of a game are unique and, together with corresponding sets A°, A^*, represent the unique full global optimal solution of a game.

2. THE MINIMAX CUBIC ALGORITHM

Suppose that the sets U, V are closed cubes with an edge of the same length $c > 0$. Then the product set $U \times V = \bar{C} \subset \mathbb{R}^n, n = n_1 + n_2$, is also a closed cube with the edge $c > 0$.

Take a point $x_0 = (u_0, v_0) \in \bar{C}$, an integer $N \ge 2$ and partition $\bar{C}$ into $N^n, n = n_1 + n_2$, equal subcubes $\bar{C}_i^1$, such that $C_i^1 \cap C_j^1 = \emptyset, i \ne j$, and $\cup \bar{C}_i^1 = \bar{C}$ (here C denotes interior of $\bar{C}$ and bar denotes the closure of C).

After partition, the point $x_0 \in \bar{C}$ will be located in one (or more, if x_0 happens to be on a common boundary) of $\bar{C}_i^1$. In any case, we assign x_0 to just one of $\bar{C}_i^1$, say, $\bar{C}_{i_0}^1$ and we call x_0 the representative of $\bar{C}_{i_0}^1$. Apply parallel translation of $\bar{C}_{i_0}^1$ to make it coincide, turn by turn, with each $\bar{C}_i^1, i = 1, 2, \ldots, N^n, i \ne i_0$; then $x_0 \in \bar{C}_{i_0}^1$ will define the representative $x_i^1 \in \bar{C}$ in each $\bar{C}_i^1$.

This rule defines the *translated grid generator* and the collection of x_i^1, $i = 1, \dots, N^n, x_{i_0}^1 = x_0$, yields the grid for any particular choice of $x_0 \in \bar{C}$. Certain subcubes $\bar{C}_i^1 \subset \bar{C}$ will be further partitioned in the same way and the upper index m of $\bar{C}_i^m, m = 1, 2, \dots$, denotes the number of partition (iteration). This procedure of partition and translated grid generation, as described in [1,2], is standard, see [5, pp. 18–23].

In accordance with (1.6), (1.7), we define for each $\bar{C}_i^m$ the consecutive deletion constants:

$$r_i^m = \alpha L_i^m d_m = \alpha L_i^m \frac{c\sqrt{n}}{N^m}, \quad L_i^m > 0, \quad m = 1, 2, \dots, \tag{2.1}$$

where $\alpha = 0.5$ if the grid point $x_i^m \in \bar{C}_i^m$ is in the center of $\bar{C}_i^m$ and $\alpha = 1$ otherwise.

Now we have to take into consideration the constituent subspaces U, V over the product cube $\bar{C} = U \times V$. Let us consider coordinates u_j^1, v_k^1 for every x_i^1; in the collection $\{x_i^1\}$ of N^n points, there are exactly N^{n_1} different $u_j^1, j = 1, \dots, N^{n_1}$, and N^{n_2} different $v_k^1, k = 1, \dots, N^{n_2}, n_1 + n_2 = n$.

ITERATION 1. Compute all $f(u_j^1, v_k^1)$ and calculate

$$s_j^1 = \max_{k \in Q} f(u_j^1, v_k^1), \quad Q = \{1, 2, \dots, N^{n_2}\}, \quad j = 1, 2, \dots, N^{n_1}. \tag{2.2}$$

Delete every $\bar{C}_i^1$ for which

$$s_j^1 - f(u_j^1, v_k^1) > r_i^1, \quad k \in Q, \quad j = 1, 2, \dots, N^{n_1}, \tag{2.3}$$

where index i corresponds to the choice of j, k, so that $(u_j^1, v_k^1) = x_i^1 \in \bar{C}_i^1$. Calculate

$$s^1 = \min_{j \in J} s_j^1, \quad J = \{1, 2, \dots, N^{n_1}\}, \tag{2.4}$$

and identify those j_0, k_0 for which $f(u_{j_0}^1, v_{k_0}^1) = s^1$. Such $(u_{j_0}^1, v_{k_0}^1) = x_{i_0}^1$ will be called *basic* points.

From the closure of remaining subcubes, delete every $\bar{C}_i^1$ for which

$$f(u_j^1, v_k^1) - s^1 > r_i^1. \tag{2.5}$$

Subcubes remaining after deletion (2.5) correspond to the index set $I_1 \subseteq I_0 = \{1, \dots, N^n\}$. The closure of those subcubes defines a quasi-cubic set:

$$\bar{K}_1 = \{x | x \in \bar{C}_i^1, i \in I_1\} \subseteq \bar{C}. \tag{2.6}$$

FURTHER ITERATIONS. Partition each $\bar{C}_i^1 \subset \bar{K}_1$ in the same way as $\bar{C}$, generate the new (finer) grid with the same translation rule and redefine the index sets $Q = \{k\}, J = \{j\}$ of (2.2), (2.4). Repeat Iteration 1 replacing u_j^1, v_k^1, s_j^1, s^1, r_i^1 by u_j^2, v_k^2, s_j^2, s^2, r_i^2 which would define $I_2, \bar{K}_2 \subseteq \bar{K}_1$. Then, partition each $\bar{C}_i^2 \subset \bar{K}_2$ in the same way and repeat Iteration 1 again, obtaining s^3, $I_3, \bar{K}_3 \subseteq \bar{K}_2$, etc.

In this process, one comes to two sequences:

$$s^1, s^2, \dots, s^m, \dots, \tag{2.7}$$

$$\bar{C} \supseteq \bar{K}_1 \supseteq \bar{K}_2 \supseteq \cdots \supseteq \bar{K}_m \supseteq \cdots. \tag{2.8}$$

Sequence (2.8) of nested compact sets has a nonempty intersection which defines its limit.

CONVERGENCE THEOREM 2.1. *Sequence (2.7) tends to a limit, and we have*

$$\lim_{m\to\infty} s^m = a^\circ = \min_{u\in U} \max_{v\in V} f(u,v), \tag{2.9}$$

$$\lim_{m\to\infty} \bar{K}_m = \bigcap_{m=1}^{\infty} \bar{K}_m = A^\circ. \tag{2.10}$$

PROOF. (a) Nonelimination of global minimaximizers.

Recall that $A^\circ \neq \emptyset$, nonempty, and let $x^\circ = (u^\circ, v^\circ) \in A^\circ$. The translated grid generator guarantees that every x_i^m, i.e., $(u_j^m,\, v_k^m)$ will remain in the iteration process until deleted with its corresponding subcube $\bar{C}_i^m$ by deletion operators (2.3) or (2.5). The variation of $f(x), x = (u,v)$, over a subcube $\bar{C}_i^m$ is bounded, cf. (1.6):

$$\operatorname{Var} f(x)|_{x\in\bar{C}_i^m} = \max_{x\in\bar{C}_i^m} f(x) - \min_{x\in\bar{C}_i^m} f(x) \le Ld^m = \frac{Lc\sqrt{n}}{N^m} = r^m. \tag{2.11}$$

This implies that for every $\bar{C}_i^m$ deleted by (2.3), we have

$$s_j^m - f(u_j^m,\, v_k^m) > r^m \ge \max_{x\in\bar{C}_i^m} f(x) - \min_{x\in\bar{C}_i^m} f(x) \ge \max_{(u,v)\in\bar{C}_i^m} f(u,v) - f(u_j^m,\, v_k^m), \tag{2.12}$$

so that within $\bar{U}_i^m \times \bar{V}_i^m = \bar{C}_i^m$, we have

$$g(u) = \max_{v\in\bar{V}_i^m} f(u,v) \le \max_{(u,v)\in\bar{C}_i^m} f(u,v) < s_j^m \tag{2.13}$$

for all $u \in \bar{U}_i^m$.

This means that within $\bar{C}_i^m$ deleted by (2.3), actually attained $\max_{v\in V_i^m} f(u,v)$ are lower than s_j^m for all $u \in \bar{U}_i^m$, so that those $\bar{C}_i^m$ cannot contain global maximizers with respect to v, thus, $\bar{C}_i^m \cap A^\circ = \emptyset$, for every $\bar{C}_i^m$ deleted by (2.3).

Furthermore, for every $\bar{C}_i^m$ deleted by (2.5), we have

$$f(u_j^m,\, v_k^m) - s^m > r^m \ge \max_{x\in\bar{C}_i^m} f(x) - \min_{x\in\bar{C}_i^m} f(x) \ge f(u_j^m,\, v_k^m) - \min_{(u,v)\in\bar{C}_i^m} f(u,v), \tag{2.14}$$

so that

$$\min_{u\in\bar{U}_i^m} g(u) = \min_{u\in\bar{U}_i^m} \max_{v\in\bar{V}_i^m} f(u,v) \ge \min_{u\in\bar{U}_i^m} \min_{v\in\bar{V}_i^m} f(u,v) = \min_{(u,v)\in\bar{C}_i^m} f(u,v) > s^m, \tag{2.15}$$

meaning that $\bar{C}_i^m$ deleted by (2.5) do not contain global minimizers of $g(u)$, that is, $\bar{C}_i^m \cap A^\circ = \emptyset$, such $\bar{C}_i^m$ not containing global minimaximizers of $f(u,v)$ over $U \times V$ (which belong to certain remaining $\bar{C}_i^m$). Denoting the intersection in (2.10) by $\bar{K}$, this proves that $A^\circ \subseteq \bar{K}$.

(b) Existence and nature of the limits.

Take any point $\tilde{x} \in \bar{K}$ and choose $x_0 = \tilde{x}$ so that $\tilde{x}$ will stay in the process indefinitely. Since $\bar{K} = \cap \bar{K}_m$, so $\tilde{x} \in \bar{K}_m$ for all $m = 1, 2, \dots$, whence $f(\tilde{x}) = f(\tilde{u}, \tilde{v}) \le s^m + r^m, \forall m$, as *remaining* after deletions (2.5). On the other hand, $f(\tilde{x}) = f(\tilde{u}, \tilde{v}) \ge s_j^m - r^m \ge s^m - r^m$, as *remaining* after deletions (2.3). Combining these two inequalities yields

$$|f(\tilde{x}) - s^m| \le r^m, \quad m = 1, 2, \dots. \tag{2.16}$$

Since $\tilde{x}$ is *fixed* and $r^m \to 0$ as $m \to \infty$, it follows from (2.16) that there exists a limit (which we denote by s°):

$$\lim_{m\to\infty} s^m = f(\tilde{x}) = s^\circ. \tag{2.17}$$

Since $\tilde{x} \in \bar{K}$ is taken arbitrarily within $\bar{K}$ and by virtue of uniqueness of the limit, we have

$$f(x) = f(u,v) = s^\circ = \text{const.} \quad \text{for all} \quad x = (u,v) \in \bar{K}. \tag{2.18}$$

Now, due to nonelimination of a global minimaximizer $x^\circ \in A^\circ \subseteq \bar{K}$ for which, by definition (1.4), $f(x^\circ) = a^\circ$, we obtain $s^\circ = a^\circ$, which proves (2.9).

To complete the proof of (2.10), it remains to verify that the set $\bar{K}$ is "clean," not containing points $x^* \in \bar{K}, f(x^*) = a^\circ$, which are *not* global minimaximizers.

If a point $x^* = (u^*, v^*) \in U \times V$ with $f(u^*, v^*) = a^\circ$ is not a global minimaximizer, then necessarily

$$f(u^*, v^*) < \max_{v \in V} f(u^*, v). \tag{2.19}$$

Suppose, on the contrary, that there is such a point $x^* \in \bar{K}$ and take $x_0 = x^*$, in which case x^* would stay in the process as representative of corresponding subcubes $\bar{C}_i^m$ for all $m = 1, 2, \ldots$. Let

$$\max_{v \in V} f(u^*, v) - f(u^*, v^*) = 2\beta > 0. \tag{2.20}$$

Since (u^*, v^*) stays in the process indefinitely, for certain $j = j(m), k = k(m)$, we have $u_{j(m)}^m = u^*, v_{k(m)}^m = v^*$ for all $m = 1, 2, \ldots$. Due to (2.1), for any $\beta > 0$, there is a number $M > 0$ such that $r^m < \beta$ for all $m > M$. The function $f(u^*, v)$ is continuous in v, hence, there is a sufficiently small subcube $\bar{C}_i^m, m > M$, with a representative point $(u^*, v_{k_*}^m) \in \bar{C}_i^m$, such that, see (2.2):

$$\max_{v \in V} f(u^*, v) \geq s_{j(m)}^m = \max_{k \in Q_{m-1}} f(u^*, v_k^m) = f(u^*, v_{k_*}^m) > \max_{v \in V} f(u^*, v) - \beta. \tag{2.21}$$

Now, from (2.20), (2.21) it follows

$$\begin{aligned} s_{j(m)}^m - f(u^*, v^*) = s_{j(m)}^m - \left[\max_{v \in V} f(u^*, v) - 2\beta\right] &= \left[s_{j(m)}^m - \max_{v \in V} f(u^*, v)\right] \\ &\quad + 2\beta > -\beta + 2\beta = \beta > r^m \end{aligned} \tag{2.22}$$

for sufficiently large $m > M$, so that the point (u^*, v^*) with its corresponding subcube would be deleted by (2.3), contradicting the choice $x^* = (u^*, v^*) \in \bar{K}$. It means that $\bar{K}$ contains only global minimaximizers, thus $\bar{K} \subseteq A^\circ$, which with the above inclusion $A^\circ \subseteq \bar{K}$ yields $\bar{K} = A^\circ$, and the proof is complete. ∎

REMARK 2.1. Deletion operators (2.3), (2.5) cannot be combined in one operator acting with respect to comparison constant s^m of (2.4) since in such a case points $x^*, f(x^*) = a^\circ$, cannot be deleted even if they are not global minimaximizers. Indeed, for $f(u,v) = u^2 - v^2$, convex-concave, we have the unique minimaximizer $u^\circ = v^\circ = 0$ in $\mathbb{R}^2$ which is actually delivered by the algorithm applied to a square containing the origin. However, if we combine (2.3), (2.5) in one operator: $|f(u_j^m, v_k^m) - s^m| > r^m$, cf. (2.16), then such an algorithm will deliver two lines $u = \pm v$ with the value $a^\circ = f(u, \pm u) = 0$, which lines are obviously not in the set $A^\circ = \{0\}$.

REMARK 2.2. It is clear that the pair $s^m, \bar{K}_m$ of (2.7), (2.8) at each iteration $m = 1, 2, \ldots$ represents an approximate global minimax solution of the game. Due to (2.11), we have $|a^\circ - s^m| \leq r^m$. Hence, if an η-precise approximate solution is sought, then iterations should continue until such m that $r^m \leq \eta$ which yields the stopping rule: the process terminates after the first $m \geq \log_N \frac{Lc\sqrt{n}}{\eta}$.

REMARK 2.3. For convenience, in Theorem 2.1 and in the stopping rule, universal constants L, r^m, cf. (2.11), are used. In practical computations, it is more efficient to use adaptive constants L_i, r_i^m as in (2.1)–(2.5).

3. THE MAXIMIN CUBIC ALGORITHM

Here, we apply the same partition and translated grid generators and the same deletion constants (2.1) as in the minimax cubic algorithm described in Section 2. We also use the same notations where applicable.

ITERATION 1. Compute all $f(u_j^1, v_k^1)$ and calculate

$$p_k^1 = \min_{j \in J} f(u_j^1, v_k^1), \quad J = \{1, 2, \ldots, N^{n_1}\}, \quad k = 1, 2, \ldots, N^{n_2}. \tag{3.1}$$

Delete every $\bar{C}_i^1$ for which

$$f(u_j^1, v_k^1) - p_k^1 > r_i^1, \quad j \in J, \quad k = 1, 2, \ldots, N^{n_2}, \tag{3.2}$$

where index i corresponds to the choice of j, k, so that $(u_j^1, v_k^1) = x_i^1 \in \bar{C}_i^1$. Calculate

$$p^1 = \max_{k \in Q} p_k^1, \quad Q = \{1, 2, \ldots, N^{n_2}\}, \tag{3.3}$$

and identify those j_0, k_0 for which $f(u_{j_0}^1, v_{k_0}^1) = p^1$(basic points). From the closure of remaining subcubes, delete every $\bar{C}_i^1$ for which

$$p^1 - f(u_j^1, v_k^1) > r_i^1. \tag{3.4}$$

Subcubes remaining after deletion (3.4) correspond to the index set $I_1^* \subseteq I_0 = \{1, \ldots, N^n\}$. The closure of those subcubes defines a quasi-cubic set

$$\bar{P}_1 = \{x | x \in \bar{C}_i^1, i \in I_1^*\} \subseteq \bar{C}. \tag{3.5}$$

FURTHER ITERATIONS. Partition each $\bar{C}_i^1 \subset \bar{P}_1$ in the same way as $\bar{C}$, generate the new (finer) grid with the same translation rule and redefine the index sets $J = \{j\}, Q = \{k\}$ of (3.1), (3.3). Repeat Iteration 1, replacing $u_j^1, v_k^1, p_k^1, p^1, r_i^1$ by $u_j^2, v_k^2, p_k^2, p^2, r_i^2$ which would define I_2^*, $\bar{P}_2 \subseteq \bar{P}_1$. Then partition each $\bar{C}_i^2 \subset \bar{P}_2$ in the same way and repeat Iteration 1 again, obtaining p^3, I_3^* $\bar{P}_3 \subseteq \bar{P}_2$, etc.

In this process, one comes to two sequences:

$$p^1, p^2, \ldots, p^m, \ldots, \tag{3.6}$$

$$\bar{C} \supseteq \bar{P}_1 \supseteq \bar{P}_2 \supseteq \cdots \supseteq \bar{P}_m \supseteq \cdots. \tag{3.7}$$

Sequence (3.7) of nested compact sets has a nonempty intersection which defines its limit.

CONVERGENCE THEOREM 3.1. *Sequence (3.6) tends to a limit, and we have*

$$\lim_{m \to \infty} p^m = a^* = \max_{v \in V} \min_{u \in U} f(u, v), \tag{3.8}$$

$$\lim_{m \to \infty}, \bar{P}_m = \bigcap_{m=1}^{\infty} \bar{P}_m = A^*. \tag{3.9}$$

Proof of this theorem follows the same lines as for Theorem 2.1 and is left to the reader as an exercise.

REMARK 3.1. Since the minimax and maximin algorithms consist of similar operations acting on the same grid, they can be combined in a unified procedure that will deliver simultaneously a°, A° and a^*, A^*. Then the difference $\Delta_m = s^m - p^m \geq 0$ evaluates the asymmetry of an approximate solution at each iteration $m = 1, 2, \ldots$. If Δ_m becomes small as $m \to \infty$, it means the existence of a saddle set (at least, approximate); otherwise, i.e., if $\lim \Delta_m = a^\circ - a^* = b > 0$, it presents a measure of the asymmetry of the game, cf. [6, Section 14.4].

EXAMPLE 3.1. Consider the game with $f(u,v) = u^2 - 4uv + v^2$, $U = [-1,1]$, $V = [-1,1]$, discussed in [3, pp. 5–6]. The computation is obvious:

$$a^\circ = \min_{u\in U}[u^2 + 4|u| + 1, \text{with } v^\circ = -\text{sgn } u] = 1 \text{ for } u^\circ = 0, \text{sgn } u = \begin{cases} 1, & u > 0 \\ \pm 1, & u = 0\,, \\ -1, & u < 0 \end{cases} \tag{3.10}$$

$$a^* = \max_{v\in V}[-3v^2, \text{ with } u^* = 2v] = 0 \text{ for } v^* = 0. \tag{3.11}$$

The game is asymmetric, $a^\circ - a^* = 1$, despite total symmetry in the data. In the brackets are indicated optimal strategies required in the case of a nonoptimal play by the opponent.

Here, we shall illustrate how to obtain the solution numerically, using the cubic algorithm for global games. Adaptive Lip constants can be determined for every subcube $\bar{C}_i^m \subset U \times V$ from the formula:

$$L_i^m = \max_{(u,v)\in \bar{C}_i^m} \|\nabla f\| = \max_{(u,v)\in \bar{C}_i^m} \sqrt{(2u-4v)^2 + (-4u+2v)^2} = 2\sqrt{5} \max_{(u,v)\in \bar{C}_i^m} \sqrt{(u-v)^2 + 0.4uv}, \tag{3.12}$$

with maximum attained on the boundary of $\bar{C}_i^m$. For the initial cube $\bar{C} = [-1,1] \times [-1,1]$, the universal constant is $L = 2\sqrt{5} \cdot \sqrt{1.9} = 8.5$ attained at $(-1,1)$ or $(1,-1)$.

In order to use the central adaptive procedures with $\alpha = 0.5$ in (2.1), see also (1.7), we take $x_0 = (0,0)$ and odd $N_1 = 5$. The values of $f(x_i^1), r_i^1$ are given within corresponding subcubes in Figure 1 and Figure 2 $\left(r_i^1 = 1.265 \max \sqrt{(u-v)^2 + 0.4uv}\right)$.

Since r_i^1 are rounded up, so equalities in (2.3), (2.5), (3.2),(3.4) delete the subcubes.

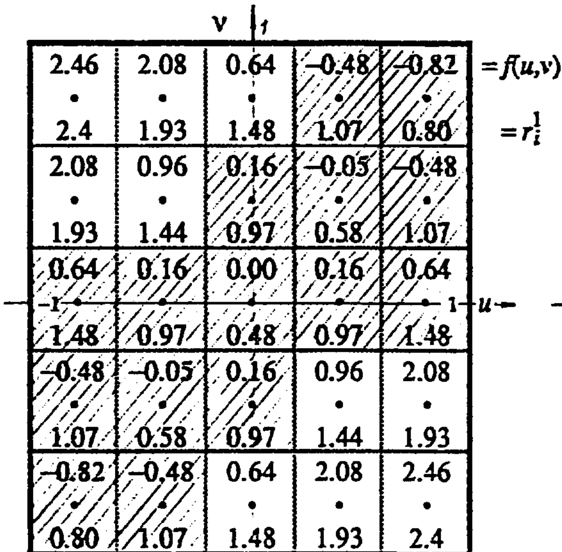

Figure 1. Minimax.

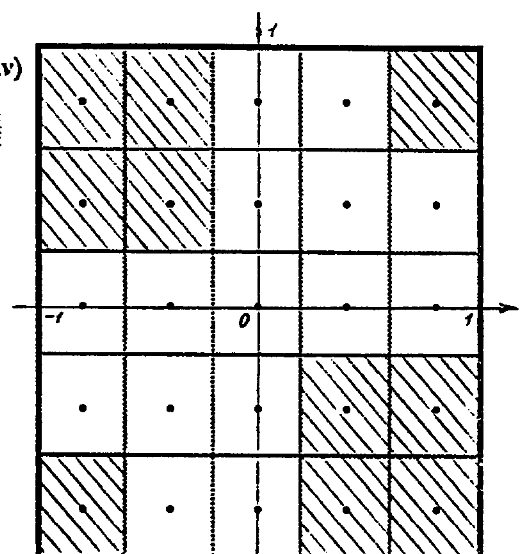

Figure 2. Maximin, same data.

Figure 1 and Figure 2 contain the same data. However, deletions (shaded areas) in Figure 1 are made according to the minimax cubic algorithm (Section 2) and in Figure 2 according to the maximin cubic algorithm (Section 3). If Player u plays first, then approximate solution of the first iteration is Figure 1: $u = 0$, $v = \pm 0.8$ with $f(u,v) = 0.64$. If Player v plays first, then approximate solution of the first iteration is Figure 2: $v = 0$, $u = 0$ with $f(u,v) = 0$ which happens to be the exact solution (but players do not know it yet, it will be known only in the limit). In further iterations, one can approach the exact solutions $u^\circ = 0$, $v^\circ = \pm 1$, $a^\circ = 1$ of (3.10), and $v^* = 0$, $u^* = 0$, $a^* = 0$ of (3.11) with any desired accuracy, and exact solutions themselves are obtained (or identified as such) in the limit.

Shaded areas show subcubes $\bar{C}_i^1$ deleted at the first iteration by (2.3), (2.5) in Figure 1, and by (3.2), (3.4) in Figure 2. We see that at the first iteration 15 of 25 subcubes (i.e., 60% of the area) are deleted in Figure 1, and 10 of 25 subcubes (i.e., 40% of the area) are deleted in Figure 2, the difference obviously depending on the cost function $f(u,v)$. This means that no function evaluations will be made within deleted areas which are, thus, excluded from further iterations.

4. PLAYING ON MISTAKES OF THE OPPONENT

For some reasons beyond the control of the players, they may not be equally capable of making the optimal choice (not equally well equipped, not equally knowledgeable, not well informed, etc.). For such a case, the players should have optimal strategies for a nonoptimal play of the opponent. From Figure 1 and Figure 2, it is easy to see that the cubic algorithm supplies such strategies provided that subcubes corresponding to extremal values s_j^m, p_k^m are also transferred to separate parallel blocks that operate by (2.2)–(2.3) or by (3.1)–(3.2) and render the values $s_j^{m_1}, p_k^{m_1}$ for the last actually performed iteration $m_1 \geq \log_N(Lc\sqrt{(n)}/\eta)$ together with corresponding basic points. Such additional blocks without deletions by (2.5), nor (3.4), can be incorporated into a combined cubic algorithm constructed according to Sections 2 and 3 to provide approximate optimal strategies for nonoptimal play of the opponent.

To illustrate the point, suppose that in Example 3.1 Player u plays $u = 0.4$. Then Player v plays $v = -0.8$ with $f(u,v) = 2.08 > a^\circ = 1$ (Figure 1). If Player v plays first with $v = 0.4$, then Player u plays $u = 0.8$ with $f(u,v) = -0.48 < a^* = 0$ (Figure 2). These choices are based, of course, on the data given by the approximation of the first iteration. In further iterations, finer grids and more accurate solutions are obtained. To get a solution not on a current grid, one may use interpolation techniques.

5. SOLUTION FOR GAMES WITH PURE CONSTRAINTS

For application of the cubic algorithm, the sets U, V in (1.4) are required to be cubes of the same length of edge $c > 0$, so that the product set $U \times V$ be a cube with diagonal $c\sqrt{n_1 + n_2}$ where n_1 and n_2 are dimensions of U and V. Generalization is straightforward for the case of box constraints $a_j \leq u_j \leq b_j, c_k \leq v_k \leq d_k$, which is suitable for differential games on classes of controls representable as linear combinations of certain functions (e.g., as partial sums of its Fourier series, or bang-bang controls, etc.).

Of much interest, however, are games with general type of constraints. Here, one can distinguish *pure* constraints that define U and V separately and independently, for example:

$$U = \{u \in \mathbb{R}^{n_1} | g_i(u) \leq 0,\ i = 1, \dots, q_1\}, \tag{5.1}$$

$$V = \{v \in \mathbb{R}^{n_2} | h_i(v) \leq 0,\ i = 1, \dots, q_2\}, \tag{5.2}$$

and *mixed* constraints that define U and V jointly, for example:

$$U \times V = \{(u,v) \in \mathbb{R}^{n_1+n_2} | \varphi_i(u,v) \leq 0,\ i = 1, \dots, q_3\}, \tag{5.3}$$

in which case, one or both players can directly affect the playing capacity of the opponent by crippling his set of feasible controls. Essential differences between those two cases can be illustrated by the following example.

EXAMPLE 5.1. Let $f(u,v) = u^3 - v^3$. With pure constraints $|u| \leq 1$, $|v| \leq 1$, the solution is $u^\circ = v^\circ = -1$, $a^\circ = \min_u \max_v f(u,v) = a^* = \max_v \min_u f(u,v) = 0$, and it is not important which player plays first. In contrast, if we add one mixed constraint $u^2 + v^2 \leq 1$, then the first player always wins and the solutions are:

$$\text{if Player } v \text{ plays first, then } v^\circ = -1, u = 0,\ a^\circ = \max_v f(0, v) = 1,$$
$$\text{if Player } u \text{ plays first, then } u^\circ = -1, v = 0,\ a^* = \min_u f(u, 0) = -1.$$

The game has become strongly asymmetric with $\Delta = a^\circ - a^* = 2$ and is won by suppressing control capabilities of the opponent. ∎

We see that there is a profound difference between games with pure and mixed constraints. In games with mixed constraints, opponents have direct influence on playing capabilities of each other; their controls and strategies are somehow coupled, dependent not through a functional but more forcibly, through a mixed constraint. In games with pure constraints, opponents cannot interfere with the choices of each other, their controls and strategies are decoupled, independent and dictated only by their respective interests defined in the cost functional. In classical literature [6,7], only games with pure constraints were considered.

For games with pure constraints, the procedures of the cubic algorithm (Sections 2 and 3) can be adjusted to provide approximate global solutions of the game in a finite number of iterations. If a set $U \times V$ is not a cube, nor a quasi-cubic set (i.e., a union of a finite number of closed cubes of equal volume, see [8], or [5, p. 82]), then a suitable procedure based on variable partition constant and on central location of grid points x_i^m (cf., the central cubic algorithm in [9] or [5, pp. 55–58]) can be described as follows.

The product set $U \times V \subset \mathbb{R}^n$ is compact, hence, bounded, and one can consider a circumscribed (not strictly) closed cube $\bar{C}, U \times V \subset \bar{C}$. Take the first representative point $x_0 \in \bar{C}$ at the center of $\bar{C}$. Take an *odd* integer $N_1 \geq 3$ and partition $\bar{C}$ into N_1^n subcubes $\bar{C}_i^1$. Apply translated grid generator as described in Section 2 to produce grid points $x_i^1 \in \bar{C}_i^1$, each x_i^1 being at the center of $\bar{C}_i^1$. Further partitions of certain $\bar{C}_i^1$ yield smaller subcubes $\bar{C}_i^m$, $m = 2, 3, \ldots$, with diameter

$$d^m = \frac{c\sqrt{n}}{N_1 N_2 \ldots N_m}, \quad m = 1, 2, \ldots, \tag{5.4}$$

where $c > 0$ is the length of edge of $\bar{C}$. Because of the central location of x_i^m, the relative variation, cf. (1.6),(1.7), has a bound:

$$\operatorname{Var} f(x) = \max_{x \in \bar{C}_i^m} \left| f(x) - f(x_i^m) \right| \leq \frac{1}{2} L d^m = \frac{Lc\sqrt{n}}{2N_1 \ldots N_m} = r^m. \tag{5.5}$$

Now the iterations are performed as described in Sections 2 and 3 with the following modifications.

(a) Comparison constant generators (2.2), (3.1) employ only those $(u_j^m, v_k^m) = x_i^m$ for which $x_i^m \in U \times V$, i.e., $u_j^m \in U$, $v_k^m \in V$.

(b) To check the membership $x_i^m \in U \times V$, one should verify that the inequalities in (5.1), (5.2) are all satisfied with u_j^m, v_k^m. If, for at least one i, we have

$$g_i(u_j^m) > \frac{L_i c\sqrt{n_1}}{2N_1 \ldots N_m}, \quad 1 \leq i \leq q_1; \quad \text{or} \quad h_i(v_k^m) > \frac{L_i^* c\sqrt{n_2}}{2N_1 \ldots N_m}, \quad 1 \leq i \leq q_2, \tag{5.6}$$

where L_i, L_i^* are Lipschitz constants of $g_i(u)$ over U and $h_i(v)$ over V respectively, then the corresponding subcube $\bar{C}_i^m$ is discarded as having empty intersection with the set $U \times V$, cf. Lemma 6.2 in [5, p. 88]. Inequalities (5.6) supply a *precise distinction operator* for the algorithm, cf. [5, pp. 83–92]. Note that at the first iteration, none of x_i^1 may be in the set $U \times V$; in this case, the process starts with exclusions by (5.6) followed by partitions.

If the set $U \times V$ is robust (i.e., $c\ell$ int $(U \times V) = c\ell(U \times V)$; a "shaved" set), then points $x_i^m \in U \times V$ necessarily appear in further iterations.

(c) If $f(u,v)$ is defined, i.e., computable, over the whole $\bar{C}$, then all remaining grid points are checked by deletion operators (2.3), (2.5), (3.2), (3.4) with r^m of (5.5) and any subcube $\bar{C}_i^m$ can be discarded irrespective of its location *vis-à-vis* the set $U \times V$. Otherwise, i.e., if $f(u,v)$ is not computable outside $U \times V$, then $x_i^m \in \bar{C}_i^m - U \times V$ are not checked by those deletion operators and corresponding $\bar{C}_i^m$ take part in further iterations.

(d) Once r^m of (5.5) becomes less than given precision $\eta > 0$, the process terminates. Then $|a^\circ - s^m| < \eta$ and $|a^* - p^m| < \eta$, hence, s^m, p^m represent the η-precise minimax and maximin values of the game. From the quasi-cubic sets $\bar{K}_m$, $\bar{P}_m$, one should discard subcubes $\bar{C}_i^m$ for which $x_i^m \notin U \times V$; then remaining subcubes provide approximate sets A°, A^* (which may contain small extra sets outside $U \times V$). Better approximations to A°, A^* are given by the intersection of all remaining subcubes with the set $U \times V$, this requiring extra computations. If just one point from A° or A^* is needed, such points are provided by basic points yielding the values s^m, p^m and delivered automatically by the procedure.

Precise distinction operator (5.6) represents a major addition to the cubic algorithms described in Sections 2 and 3. With this addition, and with selective application of generators (2.2), (3.1), the procedure becomes exact, in the limit, and resembles the beta-algorithm as described in [8] or [5, pp. 92–94]. If the sets U and V are both robust (thus, $U \times V$ is robust), then respective limits exist as $m \to \infty$, and the following result holds true.

CONVERGENCE THEOREM 5.1. *If the product set $U \times V$ is robust and subject to pure constraints only, then the following limits exist and we have*

$$\lim_{m\to\infty} s^m = a^\circ = \min_{u\in U}\max_{v\in V} f(u,v), \qquad \lim_{m\to\infty} \bar{K}_m = \bigcap_{m=1}^{\infty} \bar{K}_m = A^\circ, \tag{5.7}$$

$$\lim_{m\to\infty} p^m = a^* = \max_{v\in V}\min_{u\in U} f(u,v), \qquad \lim_{m\to\infty} = \bar{P}_m = \bigcap_{m=1}^{\infty} \bar{P}_m = A^*. \tag{5.8}$$

A proof follows the lines of Theorem 2.1 and the ideas of Theorem 6.3 in [5, pp. 95–101], as concerns nonceasing descent-ascent and the action of a precise distinction operator; it is left to the reader.

REMARK 5.1. If one of the sets U, V, say, U, is a cube $U = \bar{C}_0 \subset \mathbb{R}^{n_1}$, then one can make first a partition of C_0 into $N_1^{n_1}$ subcubes $\bar{C}_j^1 \subset \mathbb{R}^{n_1}$ of the edge $c_1 = c/N_1$, define subcubes $\bar{C}_k^1 \subset \mathbb{R}^{n_2}$ of the same edge in such a way that $V \subset \bigcup \bar{C}_k^1$ and obtain a quasi-cubic set

$$\bar{C} = \bigcup_{j,k} \bar{C}_j^1 \times \bar{C}_k^1 = \bigcup_i \bar{C}_i^1 \supset U \times V, \quad \bar{C}_i^1 = \bar{C}_j^1 \times \bar{C}_k^1. \tag{5.9}$$

Such construction eliminates many redundant subcubes that may appear in a simple enclosure $U \times V \subset \bar{C} \subset \mathbb{R}^n$ considered above, this saving computer time.

REMARK 5.2. If the product set $U \times V$ is not robust or if it is subject to mixed constraints (5.3), then not only Convergence Theorem 5.1 is invalid, but the application of the above procedure, with modifications (a)–(d) for an approximate solution, may yield wrong results due to possible elimination of all η-optimal points (minimax and/or maximin) in a case of mixed constraints or to a stop in descent-ascent for one or both sequences (2.7), (3.6), yielding some values $s^m = s^* \neq a^\circ, p^m = p^* \neq a^*$ for all $m > M$, $m \to \infty$. Thus, the above procedure is not applicable for nonrobust U or V, or for a set $U \times V$ with mixed constraints.

6. APPLICATION TO DIFFERENTIAL GAMES AND TO PURSUIT-EVASION GAMES

Consider a dynamic system with conflicting controls (a differential game):

$$\frac{dx}{dt} = f(t, x, u, v), t \in [t_0, t_f], x(t_0) = x_0 \in \mathbb{R}^s, \tag{6.1}$$

where x is the state vector, t = time, t_0 and t_f are initial and final moments and $u = u(t) \in U \subset \mathbb{R}^{n_1}, v = v(t) \in V \subset \mathbb{R}^{n_2}$ are controls with values in compact subsets U, V (this implies bounded magnitudes $\|u(t)\| \leq M_1, \|v(t)\| \leq M_2$ for all $t \in [t_0, t_f]$ which is often considered as definition of subsets U, V). Along the trajectories of (6.1), the opponents are trying to choose their controls $u(\cdot), v(\cdot)$ so as to minimize or maximize the functional

$$J(u(\cdot), v(\cdot)) = h(t_1, x[t_1]) + \int_{t_0}^{t_1} f_0(t, x[t], u(t), v(t))dt, \tag{6.2}$$

where $t_1 \leq t_f$ is appropriately defined termination moment of the game and

$$x[t] = x(t, t_0, x_0, u(\cdot), v(\cdot)), \quad x[t_0] = x_0. \tag{6.3}$$

Often there are also energy constraints on controls

$$\int_{t_0}^{t_f} \|u(t)\|dt \leq E_1, \quad \int_{t_0}^{t_f} \|v(t)\|dt \leq E_2. \tag{6.4}$$

Constraints $u \in U$, $v \in V$ as well as (6.4) are pure constraints. In contrast, terminal constraint

$$\Psi(t_1, x[t_1]) \leq 0 \tag{6.5}$$

and state variable constraints

$$x(t) \in X \subset \mathbb{R}^s, \tag{6.6}$$

which are sometimes considered may represent mixed constraints on u, v within $U \times V$.

Under standard assumptions that f and f_0 satisfy suitable conditions which guarantee unique solution of (6.1) on $[t_0, t_f]$ for all admissible control functions $u(\cdot), v(\cdot)$ with values in U, V respectively, such that the functional (6.2) takes finite values for all $t_1 \in (t_0, t_f]$, the game is well defined. The opponents then look for controls $u^\circ(\cdot), v^\circ(\cdot)$ which are the best for respective players against all controls of the opponent in the following sense:

$$J(u^\circ, v) \leq J(u^\circ, v^\circ) \leq J(u, v^\circ). \tag{6.7}$$

Such controls, if they exist, represent game-theoretic saddle point. Rules to select controls are called strategies (which may be set-valued) and rules to select *optimal* controls u°, v° of (6.7) represent optimal strategies. The value $J(u^\circ, v^\circ)$ is called the value of the game.

To determine optimal strategies, the standard approach is to use variational techniques (see, e.g., [10–14]), in particular, such as the Isaacs equation [7] or the maximum principle. For many practical problems this approach works well, especially in linear-quadratic differential games where convexity-concavity of the functional assures the unique globally optimal solution obtainable by iterative descent-ascent methods or by straightforward solution of appropriate optimality conditions. In nonlinear cases, however, even the existence of the value $J(u^\circ, v^\circ)$ is not guaranteed, see, e.g., [15]. Sufficient conditions for the application of the maximum principle or other variational techniques include, apart from usual smoothness assumptions, a bunch of other conditions difficult to check (see, e.g., [11] where nine conditions are listed). The simplest condition usually required for the existence of a saddle point (6.7) is that f, f_0, J be all separable in u, v

(recall that a function $f(z,u,v)$ is called separable in u,v, if $f(z,u,v) = f_1(z,u) + f_2(z,v)$). Another important point is whether controls $u^\circ(\cdot), v^\circ(\cdot)$ are interpreted and used as open-loop or closed-loop (feedback) controls, the latter especially useful in case of a nonoptimal play of the opponent. Optimal feedback strategies may not exist even if a saddle point exists, in which case the strategies computed, e.g., via the Isaacs equation may be invalid on the boundary of the constraint set, see [14]. Finally, if everything goes well and a variational optimal solution is obtained in the absence of the global convexity-concavity property for the problem, then it is a locally optimal solution only, and a better globally optimal solution may exist, not satisfying variational optimality conditions.

These considerations, and possible nonexistence of a value and a saddle point in (6.7), provide the motivation for development and application of direct nonvariational methods such as the cubic algorithm for the solution of differential games, to obtain the global saddle set solution (6.7), if it exists, or otherwise, the global minimax and maximin solutions which may still be useful.

6.1. Open-loop Controls

Application of the cubic (Sections 2 and 3) or beta (Section 5) algorithms to differential games requires that controls $u(t), v(t)$ be represented as linear combinations of certain basis functions

$$u(t) = \sum_{i=1}^{n_1} \alpha_i \, \varphi_i(t), \quad n_1 \le \infty, \tag{6.8}$$

$$v(t) = \sum_{i=1}^{n_2} \gamma_i \, \Psi_i(t), \quad n_2 \le \infty, \tag{6.9}$$

and that constraints on undetermined coefficients $\{\alpha_i\}, \{\gamma_i\}$ resulting from the constraints in a differential game problem be box constraints for the cubic algorithm or general *pure* constraints for the beta-algorithm.

If n_1, n_2 are finite and functions $\varphi_i(\cdot), \Psi_i(\cdot)$ are fixed according to practical feasibility and convenience for a particular dynamical system (e.g., bang-bang controls with bounded number of fixed switchings), such differential game is a game *on classes of controls.* The functions $\varphi_i(\cdot), \Psi_i(\cdot)$ may depend also on state variables, for example

$$u(t) = \sum_{i=1}^{s} \alpha_i \, x_i(t), \quad \text{or } u = \sum_{i=1}^{n_1} \alpha_i \, y_i, \quad y_i = b_i \, x(t), \tag{6.10}$$

$$v(t) = \sum_{i=1}^{s} \gamma_i \, x_i(t), \quad \text{or } v = \sum_{i=1}^{n_2} \gamma_i \, y_i^*, \quad y_i^* = b_i^* \, x(t), \tag{6.11}$$

which are linear feedbacks with full information (first formulas, all $x_i(t)$ are measured), or reduced feedbacks with incomplete information (second formulae where only linear combinations y_i, y_i^* are measured, vectors b_i, b_i^* being known). Controls (6.10)–(6.11) give a sense of feedback strategies, however, it is a restricted feedback since coefficients α_i, γ_i are fixed once for the whole game as solutions of the minimax or maximin problems. It does not make clear distinction between open-loop and closed-loop modes (see Section 6.2 below), and we call it still a game on classes of controls. Solution of such games provided by the cubic or beta algorithms is, of course, global but relative, i.e., conditioned on chosen classes of controls.

To obtain global and absolute solution, say, in $L_2(t_0, t_f)$, we have to consider (6.8)–(6.9) as Fourier series ($n_1 = n_2 = \infty$) with an appropriate choice of orthonormal bases $\{\varphi_i(\cdot)\}, \{\Psi_i(\cdot)\}$, not necessarily same for both players (e.g., trigonometric, Haar [16] or Walsh [17] systems). Then, taking partial sums with one, two, etc., terms and solving game problems of increasing dimensions, one can get a procedure resembling the delta-algorithm [5, pp. 173–223], modified accordingly for the solution of continuous games. In this way, the full global optimal solutions (minimax and/or maximin) can be obtained in $L_2(t_0, t_f)$, exact in the limit (for a convergent Fourier series) or with a given precision, in a finite number of iterations. Since any physically sound control function can be represented as a Fourier series, it is clear that expansions (6.8)–(6.9) in $L_2(t_0, t_f)$ provide the global open-loop solution of all practical differential game problems. This solution, of course, will be approximate; however, in realistic situations, differential game problems admitting exact variational optimality conditions are usually solved also approximately, see, e.g., [14, p. 131, Theorem 3.4]. Due to imprecision of mathematical models, their exact solutions, if any, are approximate ones from the physical point of view.

To solve the problem numerically via the cubic or beta algorithms (Sections 2, 3 and 5), we observe that the only requirements for the application of the algorithms are that the function $f(u_j^m, v_k^m)$ be computable for each pair $(u_j^m, v_k^m) \in U \times V$, the Lip constraints L_i be somehow estimated for each subcube $\bar{C}_i^m$, the constraints be *pure* and the product set $U \times V$ be *robust*. With controls $u(\cdot)$, $v(\cdot)$ represented by (6.8)–(6.9) or (6.10)–(6.11), the role of u_j^m, v_k^m is played by points $(\alpha_j^m, \gamma_k^m) = x_i^m \in \bar{C}_i^m$, where $\alpha = (\alpha_1, \dots, \alpha_{n_1}) \in U' \subset \mathbb{R}^{n_1}, \gamma = (\gamma_1, \dots, \gamma_{n_2}) \in V' \subset \mathbb{R}^{n_2}$ are vector parameters to yield the solutions

$$a^\circ = f(\alpha^\circ, \gamma^\circ) = J(u^\circ(\cdot), v^\circ(\cdot)) = \min_{u \in U} \max_{v \in V} J(u, v) \tag{6.12}$$

$$a^* = f(\alpha^*, \gamma^*) = J(u^*(\cdot), v^*(\cdot)) = \max_{v \in V} \min_{u \in U} J(u, v). \tag{6.13}$$

To compute $J(u, v)$ of (6.2) for each (α_j^m, γ_k^m), we use the correspondence $u(t) \Leftrightarrow \alpha$, $v(t) \Leftrightarrow \gamma$ given in the control representations (6.8)–(6.9),(6.10)–(6.11), putting $u(t)$, $v(t)$ into the system (6.1) to compute the functional (6.2). The product set $U' \times V'$ for (α, γ) is induced by the original constraint sets $U \ni u(t), V \ni v(t)$ and other constraints such as (6.4), or by a choice of the class of functions with appropriate sets U', V' for parameters α, γ subject, possibly, to other constraints as (6.4). For pursuit-evasion games with energy constraints and many other differential game problems, resulting sets U', V' for α, γ will be robust and independent so that the product set $U' \times V'$ will be robust and formed by pure constraints. With values of the functional $J = f(\alpha_j^m, \gamma_k^m)$ already computed, the Lip constants L_i can be readily estimated, e.g., by the procedure described in [5, pp. 69–78] (of course, functions $h(\cdot), f_0(\cdot)$ in (6.2) should be Lipschitzian).

6.2. Closed-loop Strategies

Here, we propose a method to determine approximations to globally optimal closed-loop strategies for a particular class of differential games with pure constraints. We shall use subdivisions of the time interval $[t_0, t_f) = \bigcup [t_i, t_{i+1}), i = 0, 1, \dots, n-1, t_n = t_f$, and piecewise constant controls $u(\cdot), v(\cdot)$, constant over every subinterval $[t_i, t_{i+1})$. Thus, parameters α, γ will be just values of controls $u(t) = \alpha, v(t) = \gamma, t \in [t_i, t_{i+1})$, for some fixed subinterval $i = i', 0 \le i' < n, \alpha \in U' \subset \mathbb{R}^{n_1}, \gamma \in V' \subset \mathbb{R}^{n_2}$, where n_1, n_2 are numbers of control functions for each player. No separability nor convexity assumptions are imposed. However, we consider only such differential games with pure constraints that satisfy the following condition.

HYPOTHESIS. There exists a finite subdivision $[t_0, t_1) \cup [t_1, t_2) \cup \dots \cup [t_{n-1}, t_f) = [t_0, t_f)$ such that for this subdivision and for every finer subdivision obtained by partition of intervals of the original subdivision, the optimality of the game over each subinterval implies the optimality of the game over the whole interval $[t_0, t_f)$.

Here, we use the term "optimality" in a broad sense to mean minimax or maximin for respective players. This hypothesis means the absence of singularities in some region containing globally optimal solutions and holds in many practical cases of pursuit-evasion games and other differential games. Numerically, it allows us to obtain globally optimal closed-loop strategies step-by-step with the help of the beta-algorithm that can be used as on-line procedure to utilize possible mistakes of the opponent. If the game is stationary, that is, f in (6.1), f_0 in (6.2) and constraints do not depend explicitly on time, then the solution can be done once for the first subinterval $[t_0, t_1)$ and applied to subsequent states $x[t_1], x[t_2], \dots$, considered as new initial conditions. Over a single subinterval $[t_i, t_{i+1}]$, the solution follows the procedures described in Sections 2, 3 and 5 with the computation of $J = f(\alpha, \gamma)$ as in Section 6.1 above for a new parametrization of piecewise constant controls $u(\cdot), v(\cdot)$. By taking sufficiently fine subdivisions, one can obtain accurate approximations to globally optimal closed-loop strategies. Since the cubic and beta algorithms deliver full global optimal solutions, those approximations will be set-valued (quasi-cubic sets of equioptimal parameters α, γ).

6.3. Example

Consider the popular homicidal chauffeur model for pursuit-evasion [7, 18–20]. A detailed variational solution of this problem with variable speed of pursuer and other important generalizations can be found in [19]. With rather modest intention of illustrating the numerical procedure of constructing the closed-loop minimax and maximin strategies via cubic algorithm, we take the simple classical problem from [7,18], and we reproduce its statement as given in [18, pp. 292–293].

"Problem 4. (See Isaacs, pp. 28, 273). In a planar minimax intercept-time problem the pursuer and the evader have constant velocity magnitudes, V_p and V_e, respectively ($V_p > V_e$). The evader has direct control of the direction of his velocity, whereas the pursuer has control over his lateral acceleration, which is bounded. The pursuer, thus, has a minimum radius of turn, R.

In a coordinate frame centered in the pursuer (see Figure 3) with y-axis always kept parallel to the pursuer's velocity vector, the relative position of the evader is (x, y), where

$$\dot{x} = V_e \sin\Psi - V_p \frac{y}{R} u, \quad \dot{y} = V_e \cos\Psi - V_p + V_p \frac{x}{R} u; \quad -1 \le u \le 1. \tag{A1}$$

Here, Ψ is the evader's control (unbounded) and u is the pursuer's control (bounded, $|u| \le 1$). The intercept time t_f is determined by

$$(x^2 + y^2)_{t=t_f} = \ell^2, \tag{A2}$$

and the initial conditions $x(0)$, $y(0)$ are specified. Show that the minimax strategies (under some conditions) are

$$u = -\mathrm{sgn}(\theta - \Psi), \quad \text{where } \theta = \tan^{-1}\frac{x}{y}; \ \dot{\Psi} = \frac{V_p}{R}\,\mathrm{sgn}\,(\theta - \Psi)."\tag{A3}$$

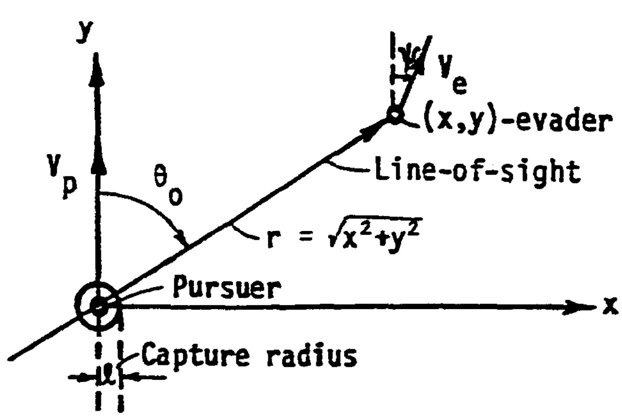

Figure 3. Nomenclature to Problem 4.

Derivation of the minimax strategies (A3) under suitable conditions can be found in [19, pp. 81–87]. If we assume that direct control of the velocity of the evader, as stated above, means that he can change the direction of V_e instantaneously, or at least with a speed not less than

$$\left|\frac{d\Psi}{dt}\right| \geq \left|\frac{\dot{x}y - x\dot{y}}{x^2+y^2}\right| = V_p\left|\frac{x}{x^2+y^2} - \frac{u}{R}\right|, \tag{6.14}$$

then the solution is very simple.

Consider the Lyapunov function [20],

$$V = \frac{1}{2}(x^2+y^2), \tag{6.15}$$

with its total derivative on the trajectories of (A1):

$$\dot{V} = \frac{dV}{dt} = (x\sin\Psi + y\cos\Psi)V_e - yV_p, \tag{6.16}$$

which does not depend on pursuer's control u. It is obvious that evasion occurs if $\dot{V} \geq 0$; capture is achieved if $\dot{V} < 0$, at least asymptotically, and in a finite time t_f if $\dot{V} \leq \alpha < 0$. Taking partial derivative and equating it to zero,

$$\frac{\partial\dot{V}}{\partial\Psi} = x\cos\Psi - y\sin\Psi = 0, \tag{6.17}$$

we find the optimal feedback strategy

$$\Psi^\circ = \arctan\frac{x}{y} \tag{6.18}$$

that delivers maximum to $\dot{V}$ (since $\frac{\partial^2\dot{V}}{\partial\Psi^2} = -\frac{2V}{y}\cos\Psi < 0$) irrespective of pursuer's control u.

Substituting (6.18) into (6.16), we get

$$\dot{V}(\Psi^\circ) = V_e\sqrt{x^2+y^2} - yVp \geq 0, \tag{6.19}$$

if and only if

$$\frac{y}{\sqrt{x^2+y^2}} = \cos\theta \leq \frac{V_e}{V_p} < 1. \tag{6.20}$$

Denoting the angle $\theta_0 =$ $\arccos(V_e/V_p)$, indicated in Figure 3, we obtain that in the angular region

$$\frac{\pi}{2} \geq \theta \geq \theta_0 = \arccos\frac{V_e}{V_p}, \tag{6.21}$$

evasion is guaranteed by the control Ψ°, (6.18), irrespective of pursuer's control u.

If initial conditions $x(0), y(0)$ are within the possible capture region

$$\theta_0 > \theta = \text{arc tan}\ \frac{x}{y} \geq 0, \tag{6.22}$$

then, whatever the distance $r(t) = \sqrt{x^2+y^2}$, the pursuer should choose the control $u^\circ = +1$ (cf. with $u = -\text{sgn}(\theta - \Psi)$ in (A3) and with $u = -1$ in [19, p. 87]) in order to increase y and keep $\dot{V}(\Psi^\circ) < 0$ in (6.19) (from (A1) and (6.19), one may see that $u = +1$ brings an overall decrease to $\dot{V}(\Psi^\circ)$). Thus, if the evader can keep the control (6.18), then the pair $\Psi^\circ, u^\circ = +1$ represents the minimax intercept-time feedback solution of the problem. Substituting Ψ°, u° into (A1) and solving the resulting *nonlinear* system, one can determine the capture time t_f, if it exists, from the equation (A2).

For a big ship, manoeuvrability condition (6.14) does not hold and the above analysis may be invalid. Let us consider a more realistic situation of a ship and a torpedo hitting the water at some distance from the ship in a position in which capture of the ship is possible. Let us take $R = 1$ (or dimensionless variables as in [19]), $V_e = 1, V_p = 2$, so that the capture angle is

$$0 \le \theta < \theta_0 = \arccos \frac{V_e}{V_p} = \arccos \frac{1}{2} = \frac{\pi}{3} = 1.05. \tag{6.23}$$

Take $t_0 = 0$ and assume that initial position of the ship and a torpedo is such that $x(0) = y(0) = 1$, so that $\theta(0) = \arctan \frac{x(0)}{y(0)} = \frac{\pi}{4} = 0.785 < \theta_0$ which corresponds to the capture situation. In accordance with the problem and Figure 3, sets of control values are $u \in [-1, 1], \Psi \in [0, \frac{\pi}{2}]$. Suppose that initial positions of rudders of the ship and torpedo are such that $u(0) = 0$ and $\Psi(0) = \frac{\pi}{8} = 0.4$, that is, they are *not* in the optimal positions $u^\circ(0) = +1, \Psi^\circ(0) = \frac{\pi}{4} = 0.785 = \theta(0)$.

With $R = 1, V_e = 1, V_p = 2$, equations (A1) have the following discretization:

$$\Delta x = (\sin \Psi - 2yu)\Delta t, \qquad x(0) = 1, t \ge 0 \tag{6.24}$$

$$\Delta y = (\cos \Psi - 2 + 2xu)\Delta t, \quad y(0) = 1. \tag{6.25}$$

Take $\Delta t = 0.1$ and assume manoeuvrability of the ship $|\Delta\Psi| \le 0.1$ and of the torpedo $|\Delta u| \le 0.2$ during the incremental time $\Delta t = 0.1$. In accordance with (A2), the cost function over each period $\Delta t = 0.1$ is the squared distance $r^2(t) = x^2 + y^2 = f(u, \Psi)$ with initial value $r^2(0) = 2$. We do not fix the capture distance ℓ^2 which defines the termination moment t_f, but does not affect the optimal closed-loop strategies.

The application of the cubic or beta algorithms, of course, requires a computer. In order not to just present computational results for the reader to trust, but to exhibit the procedure for the reader to use, and to demonstrate the applicability of the algorithms for the feedback minimax solution of a differential game with possible utilization of mistakes of the opponent, we need simplified transparent calculations that the reader could repeat himself. To this end, we use the simpler cubic algorithm; at each $t_n = n \cdot \Delta t, n = 1, 2, \ldots$, we make only one, first, iteration, choosing $s_n^1, p_n^1, n = 1, 2, \ldots$, and corresponding basic points (controls) u_n, Ψ_n without further partitions (thus, we do not need deletion operators to decrease the number of function evaluations); subdivisions are made with $N = 3$ and grid points are chosen at convenience (no translated grid generator) without attempts to make a local improvement or grid refinement.

The problem has simple box constraints, but nonseparable cost function $f(u, \Psi)$; it seems also that it satisfies the Hypothesis in Section 6.2, though we cannot verify it. We shall use piecewise constant controls in the following procedure.

According to the manoeuvrability constraints $|\Delta\Psi| \le 0.1, |\Delta u| \le 0.2$ and initial conditions for controls: $\Psi(0) = 0.4, u(0) = 0$, we have for the period $t \in [0, 0.1]$ the following constraint sets: $\Psi \in [0.3, 0.5], u \in [-0.2, 0.2]$. Subdividing those segments in three subsegments and taking grid points $\Psi_{1,2,3} = \{0.3, 0.4, 0.5\}, u_{1,2,3} = \{-0.2, 0, 0.2\}$, we obtain nine combinations $x_i^1 = (\Psi_j, u_k), (j, k = 1, 2, 3;\ i = 1, \ldots, 9)$; note that, due to the absence of deletions and partitions, we do not need the "cubic" configuration of the product subsets represented by the points x_i^1.

Substituting controls Ψ_j, u_k into (6.24),(6.25) with $x(0) = y(0) = 1, \Delta t = 0.1$, we calculate Δx_1, Δy_1, and the values $x_1 = x(0) + \Delta x_1$, $y_1 = y(0) + \Delta y_1, r_1^2 = x_1^2 + y_1^2$ for all nine values of (j, k), see Figure 4. From the nine values of the cost function r_1^2, we select as usual $\min_u \max_\Psi r_1^2, \max_\Psi \min_u r_1^2$, which define optimal controls $\Psi_1^\circ = 0.5, u_1^\circ = 0.2$ on $[0, 0.1]$ and resulting states $x(0.1) = 1.01, y(0.1) = 0.93$. In our case, we have a saddle point with the value $r_1^2 = 1.885$ (bold square in Figure 4).

Now, for the period $t \in (0.1, 0.2]$, we have the constraint sets $\Psi \in [0.4, 0.6]$, $u \in [0, 0.4]$ within which we take controls as shown in Figure 5. Repeating the first iteration again from the new

u \ ψ	0.3	0.4	0.5
−0.2	1.885	1.889	1.916
0	1.871	1.874	1.895
0.2	1.864	1.865	1.885

$x(0.1) = 1.01; y(0.1) = 0.93$

$\theta(0.1) = 0.827 > \psi_1^{\bullet} = 0.5$

Figure 4. Period $t \in [0, 0.1]$.

u \ ψ	0.4	0.5	0.6
0	1.775	1.796	1.801
0.2	1.760	1.780	1.783
0.4	1.770	1.770	1.772

$x(0.2) = 0.99; y(0.2) = 0.89$

$\theta(0.2) = 0.839 > \psi_2^{\bullet} = 0.6$

Figure 5. Period $t \in (0.1, 0.2]$.

initial position $x(0.1) = 1.01$, $y(0.1) = 0.93$, we obtain optimal controls $\Psi_2^\circ = 0.6, u_2^\circ = 0.4$, the saddle point value $r_2^2 = 1.772$ and the new position $x(0.2) = 0.99$, $y(0.2) = 0.89$. It is worth noting that evader's control angle $\Psi^\circ(t)$ is increasing to approach the line of sight, $\Psi^\circ(t) \to \theta(t)$, which is theoretically optimal according to (6.18).

To see the work of the algorithm, let us skip the optimization procedure in the third and fourth runs, simply adding the maximum increments to control values by analogy with Figure 4 and Figure 5 until after such moment $t*$ that $\Psi^\circ(t*) > \theta(t*)$. In the third run, this yields $\Psi_3^\circ = 0.7$, $u_3^\circ = 0.6$, $r_3^2 = 1.686$, $x(0.3) = 0.95$, $y(0.3) = 0.885$, still $\Psi_3^\circ < \theta(0.3) = 0.821$. In the fourth run, we have $\Psi_4^\circ = 0.8$, $u_4^\circ = 0.8$, $r_4^2 = 1.597$, $x(0.4) = 0.88$, $y(0.4) = 0.907$, and $\Psi_4^\circ > \theta(0.4) = 0.770$, so we make another iteration in the fifth run with $\Psi \in [0.7, 0.9], u \in [0.6, 1]$, see Figure 6.

u \ ψ	0.7	0.8	0.9
0.6	1.488	1.489	1.486
0.8	1.492	1.494	1.489
1.0	1.504	1.503	1.497

$x(0.5) = 0.84; y(0.5) = 0.88$

$\theta(0.5) = 0.762 < \psi_5^{\bullet} = 0.8$

Figure 6. Period $t \in (0.4, 0.5]$.

In the fifth run, Figure 6, we again have the saddle point value $r_5^2 = 1.489$, but optimal controls are not at their maxima on the right boundary of respective constraint sets as in Figure 4 and Figure 5. We have $\Psi_5^\circ = 0.8$ in the interior of $[0.7, 0.9]$ and $u_5^\circ = 0.6$ switched to the left boundary of its constraint set $[0.6, 1.0]$. This means that the strategies of the right boundary values that proved optimal in the first two runs, Figure 4 and Figure 5, and were extrapolated in the third and fourth runs, are no more optimal; the solution presented in Figure 6 may be too rough and the application of the full procedure with deletions and grid refinements in further iterations of the cubic algorithm is necessary to obtain better closed-loop strategies and a smaller value r_5^2 at $t = 0.5$. In this way, one continues successive runs of the cubic algorithm for $n = 6, 7, \ldots$, until the capture occurs with the fulfilment of (A2) or the evader enters the safe zone (6.21) and wins. (Note that the evader may concentrate his efforts not on maximizing the distance $r^2(t)$, but on the entry into the safe zone (6.21) before the capture (A2), with the pursuer trying to prevent him from doing so. This would be a different game perfectly solvable with the cubic algorithm by simple replacement of the cost function, that is of the values $r^2(t)$ in tables in Figure 4 to Figure 6 with the values of $\theta(t) = \text{arc tan } (x/y)$ keeping in check conditions (A2) and (6.21) for termination of the game).

Let us consider our solution *vis-à-vis* the solution (A3). Strategies (A3) presume manoeuvrability of the opponents within the period $\Delta t = 0.1$ in the range $|\Delta\Psi| \leq 2(V_p/R)\Delta t = 0.4$ for $V_p = 2, R = 1$, and $|\Delta u| \leq 2$ which is four and ten times more than our bounds of $|\Delta\Psi| \leq 0.1, |\Delta u| \leq 0.2$. It means that two solutions are not comparable. Obviously, our solution (even if exact) is not optimal for more manoeuvrable crafts. However, solution (A3) is not applicable to less manoeuvrable crafts. Variational solution produced without manoeuvrability constraints may prove not applicable in a real situation. Manoeuvrability constraints, however, imply nonstationary constraint sets for admissible values of controls. Introduction of variable constraint sets for controls will complicate the variational solution and also the solution by the cubic algorithm in the case of open-loop strategies. But in the case of closed-loop strategies constructed via cubic algorithm, this has absolutely no importance and constraint sets may depend on time and on state variables.

7. CONCLUSIONS

In this paper, we consider nonconvex continuous and differential games. No separability nor convexity-concavity assumptions are imposed on the cost functional which may be also nonsmooth. In such setting, the value of the game and saddle points generally do not exist and, if they do, gradient or variational methods may deliver only local solutions. We do not use here those methods nor Lagrangian formalism, and we do not assume saddle point values to exist. Instead, we look for the unique global minimax and maximin values for both players, and for the full sets of controls that deliver those values. If it happens that minimax = maximin, then the algorithms deliver the global value of the game and the entire global saddle set. To solve the problem, global set-to-set descent-ascent methods are used with distinction and rejection of whole sets not containing extremal points.

First, the minimax and maximin cubic algorithms are proposed for solution of nonlinear Lipschitz continuous games defined over a cube with nonseparable, non-convex-concave cost function. Convergence theorems prove that the algorithms deliver, in the limit, the exact unique full global optimal solution of the game in the above sense. It follows that an approximate solution up to any desired accuracy can be found in a finite number of iterations; a stopping rule depending on desired precision is provided for this purpose. Two algorithms are then combined in one and illustrated by an example with no saddle point. It is demonstrated that the algorithms with minor addition also supply strategies for playing on mistakes of the opponent.

Second, the combined algorithm is modified and supplied with special distinction operator for full global optimal solution of nonconvex Lipschitz continuous games defined over arbitrary compact and robust set formed by pure constraints (distinction between pure and mixed constraints is made and illustrated by an example). Convergence theorem is given with brief outline of the proof.

Third, the results are applied to differential games and, in particular, to pursuit-evasion games. After a short discussion, we show how to apply the algorithms for full global open-loop solution of a differential game on specified classes of controls (that may be practically convenient) or on subsets of general functional spaces with a basis. We consider, specifically, the representation by a Fourier series in $L_2(t_0, t_f)$, looking for approximate solutions. In this case, compactness or robustness of control sets are not required; we need, however, to translate control set specifications into pure constraints on coefficients of successive partial sums.

Finally, we consider application of the algorithms to construction of the globally optimal closed-loop strategies for both players in an example of the ship-torpedo collision-avoidance game (the popular homicidal chauffeur model). We consider a realistic situation with manoeuvrability restrictions on control systems and with some initial position of rudders at the moment torpedo hits the water. The procedure is illustrated in detail, and a saddle point appears without any preconditions. Certain important features are revealed, such as the possibility to handle non-stationary, time and state-dependent constraints on control values, i.e., the sets $U(t,x), V(t,x)$, without any complications; the possibility of obtaining suboptimal solutions to avoid excessive partitions and save computing time; and the possibility to replace cost functionals within the iteration process according to changing interests of the players.

REFERENCES

1. E.A. Galperin, The cubic algorithm, Preliminary report, *Abstracts of the American Mathematical Society,* *274* (6), p. 501 (October 1983).
2. E.A. Galperin, The cubic algorithm, *Journal of Mathematical Analysis and Applications* **112**, 635–640 (1985).
3. E.A. Galperin and Q. Zheng, *New Theory of Continuous Games*, NP Research Publ., Montreal, Quebec, Canada, (1990).
4. E.A. Galperin and Q. Zheng, *Global Solutions in Optimal Control and Games*, NP Research Publ., Montreal, Quebec, Canada, (1991).
5. E.A. Galperin, *The Cubic Algorithm for Optimization and Control*, NP Research Publ., Montreal, Quebec, Canada, (1990).
6. J. von Neumann and O. Morgenstern, *Theory of Games and Economic Behavior*, Princeton University Press, Princeton, (1953).
7. R. Isaacs, *Differential Games*, Wiley, New York, (1965).
8. E.A. Galperin, The beta-algorithm, *Journal of Mathematical Analysis and Applications* **126**, 455–468 (1987).
9. E.A. Galperin, Precision, complexity and computational schemes of the cubic algorithm, *Journal of Optimization Theory and Applications* **57**, 223–238 (1988).
10. L.D. Berkovitz, A variational approach to differential games, In *Advances in Game Theory, Ann. of Math. Studies*, 127–175, (1964).
11. L.S. Pontryagin, To the theory of differential games, *Uspekhi Math. Nauk* (in Russian) **21** (4 (140)) (1966).
12. N. Krassovski and A. Soubbotine, *Jeux Differentiels*, Mir, Moscou, (1977).
13. Y.C. Ho, A.E. Bryson and S. Baron, Differential games and optimal pursuit-evasion strategies, *IEEE Trans. on Automatic Control* **16** (4), 385–389 (1965).
14. M. Heymann, N. Rajan and M. Ardema, Optimal strategies in event-constrained differential games, *Computers Math. Applic., Pursuit-evasion Differential Games II* (Edited by Y. Yavin and M. Pachter) **18** (1–3), 127–131 (1989).
15. L.D. Berkovitz, A differential game without pure strategy solutions on the open set, In *Advances in Game Theory, Ann. of Math. Studies*, 175–195, (1964).
16. A. Haar, Zur Theorie der orthogonalen Funktionensysteme, *Math. Ann.* **69**, 331–371 (1910).
17. J.L. Walsh, A property of Haar's system of orthonormal functions, *Math. Ann.* **90**, 38–45 (1923).
18. A.E. Bryson, Jr. and Y.-C. Ho, *Applied Optimal Control*, (1975).
19. T. Miloh and M. Pachter, Ship collision-avoidance and pursuit-evasion differential games with speed-loss in a turn, *Computers Math. Applic., Pursuit-evasion Differential Games II* (Edited by Y. Yavin and M. Pachter) **18** (1–3), 77–100 (1989).
20. E.A. Galperin and J.M. Skowronski, Playable asymptotic observers for differential games with incomplete information—the user's guide, In *Proc. of the 23rd Conference on Decision and Control*, Las Vegas, NV, pp. 1201–1206, (December 1984).